

What is claimed is:

1 1. A system for actively managing an enterprise of configurable
2 components, comprising:
3 at least one individual component in a managed domain storing a
4 configuration comprising a set of configuration parameters and type definitions;
5 at least one document type definition defining a mapping of the
6 configurations between at least two of the individual components, each specifying
7 a configuration parameter with a relationship dependent on at least one other such
8 configuration parameter;
9 a management system registering each such individual component,
10 validating the configuration parameters and type definitions based on a master set
11 of configuration parameters and type definitions defined for an enterprise, and
12 enforcing the mappings within the enterprise by dynamically probing the
13 validated configurable parameters for each such individual component.

1 2. A system according to Claim 1, further comprising:
2 a service interface discovering a change in at least one such configuration
3 parameter through dynamic probing.

1 3. A system according to Claim 2, further comprising:
2 a set of core services taking action on the discovered change, comprising
3 at least one of:
4 a service changing at least one such configuration parameter back
5 to a default or previously validated value;
6 a service auditing the at least one such configuration parameter;
7 a service generating an alert about the at least one such
8 configuration parameter; and
9 a service acquiescing to the discovered change.

1 4. A system according to Claim 2, further comprising:
2 a log storing the probed change in the at least one such configuration
3 parameter.

1 5. A system according to Claim 1, further comprising:
2 a configuration and validation module performing an impact analysis on
3 the enterprise against at least one such configuration parameter.

1 6. A system according to Claim 1, wherein the relationship comprises
2 one of one-way, two-way, cyclic, one-to-many, many-to-one, and many-to-many.

1 7. A method for actively managing an enterprise of configurable
2 components, comprising:
3 storing a configuration for at least one individual component in a managed
4 domain, comprising a set of configuration parameters and type definitions;
5 defining a mapping of the configurations between at least two of the
6 individual components, each specifying a configuration parameter with a
7 relationship dependent on at least one other such configuration parameter;
8 registering each such individual component and validating the
9 configuration parameters and type definitions based on a master set of
10 configuration parameters and type definitions defined for an enterprise; and
11 enforcing the mappings within the enterprise by dynamically probing the
12 validated configurable parameters for each such individual component.

1 8. A method according to Claim 7, further comprising:
2 discovering a change in at least one such configuration parameter through
3 dynamic probing.

1 9. A method according to Claim 8, further comprising:
2 taking action on the discovered change, comprising at least one of:
3 changing at least one such configuration parameter back to a
4 default or previously validated value;
5 auditing the at least one such configuration parameter;
6 generating an alert about the at least one such configuration
7 parameter; and

8 acquiescing to the discovered change.

1 10. A method according to Claim 8, further comprising:
2 logging the probed change in the at least one such configuration
3 parameter.

1 11. A method according to Claim 7, further comprising:
2 performing an impact analysis on the enterprise against at least one such
3 configuration parameter.

1 12. A method according to Claim 7, wherein the relationship
2 comprises one of one-way, two-way, cyclic, one-to-many, many-to-one, and
3 many-to-many.

1 13. A computer-readable storage medium holding code for performing
2 the method according to Claims 7, 8, 9, 10, 11, or 12.

1 14. A system for providing a framework for centrally managing
2 configurations of distributed computing components, comprising:
3 a plurality of individual components each comprising a client module
4 applying document type definitions storing configuration parameters;
5 a management server maintaining a database repository storing master
6 document type definitions and global parameter definitions; and
7 a management console accessing the database repository and comprising:
8 at least one service interface retrieving the stored document type
9 definitions for each individual component via the corresponding client module;
10 a parser extracting the configuration parameters from each
11 retrieved document type definition with the master document type definitions; and
12 a validator validating each extracted configuration parameter
13 against the validated configuration parameters in the global parameter definitions.

1 15. A system according to Claim 14, further comprising:

2 at least one adapter accessing component-specific configuration
3 parameters maintained with at least one such individual component.

1 16. A system according to Claim 14, further comprising:
2 at least one component-specific adapter dynamically probing the
3 individual components.

1 17. A system according to Claim 14, further comprising:
2 a component parameter relationship dependency tree formed from the
3 extracted configuration parameters; and
4 an impact analyzer determining propagated changes by traversing the
5 component parameter relationship dependency tree.

1 18. A system according to Claim 14, further comprising:
2 a change manager effecting a change to a configuration parameter with at
3 least one such individual component.

1 19. A system according to Claim 14, further comprising:
2 a set of doclets formed from the extracted configuration parameters.

1 20. A system according to Claim 19, wherein the doclets are written in
2 a scripting language comprising the Extensible Markup Language.

1 21. A system according to Claim 14, further comprising:
2 validation services, comprising at least one of:
3 managing at least one such configuration parameter;
4 advising about at least one such configuration parameter;
5 alerting about at least one such configuration parameter; and
6 acquiescing to at least one such configuration parameter.

1 22. A system according to Claim 14, further comprising:
2 a browsing service providing a user interface management console.

1 23. A system according to Claim 14, further comprising:
2 a management configuration module registering new such individual
3 components.

1 24. A system according to Claim 14, further comprising:
2 a management configuration module updating changed such individual
3 components.

1 25. A system according to Claim 14, wherein the individual
2 components comprise at least one of a Web server, an internet application server
3 and a database server.

1 26. A method for providing a framework for centrally managing
2 configurations of distributed computing components, comprising:
3 interfacing to a plurality of individual components each comprising a
4 client module applying document type definitions and storing configuration
5 parameters;
6 maintaining a database repository storing master document type
7 definitions and global parameter definitions; and
8 accessing the database repository and actively managing the individual
9 components, comprising:
10 retrieving the stored document type definitions for each individual
11 component via the corresponding client module;
12 extracting the configuration parameters from each retrieved
13 document type definition with the master document type definitions; and
14 validating each extracted configuration parameter against the
15 validated configuration parameters in the global parameter definitions.

1 27. A method according to Claim 26, further comprising:
2 accessing component-specific configuration parameters maintained with at
3 least one such individual component.

1 28. A method according to Claim 26, further comprising:
2 dynamically probing the individual components.

1 29. A method according to Claim 26, further comprising:
2 forming a component parameter relationship dependency tree from the
3 extracted configuration parameters; and
4 determining propagated changes by traversing the component parameter
5 relationship dependency tree.

1 30. A method according to Claim 26, further comprising:
2 effecting a change to a configuration parameter with at least one such
3 individual component.

1 31. A method according to Claim 26, further comprising:
2 forming a set of doclets formed the extracted configuration parameters.

1 32. A method according to Claim 31, wherein the doclets are written in
2 a scripting language comprising the Extensible Markup Language.

1 33. A method according to Claim 26, further comprising:
2 providing validation services, comprising at least one of:
3 managing at least one such configuration parameter;
4 advising about at least one such configuration parameter;
5 alerting about at least one such configuration parameter; and
6 acquiescing to at least one such configuration parameter.

1 34. A method according to Claim 26, further comprising:
2 providing a browsing service comprising a user interface management
3 console.

1 35. A method according to Claim 26, further comprising:
2 registering new such individual components.

1 36. A method according to Claim 26, further comprising:
2 updating changed such individual components.

1 37. A method according to Claim 26, wherein the individual
2 components comprise at least one of a Web server, an internet application server
3 and a database server.

1 38. A computer-readable storage medium holding code for performing
2 the method according to Claims 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, or 37.